Fuzzy Allocation of Fine-Grained Compute Resources for Grid Data Streaming Applications

Wen Zhang, Tsinghua University, China
Junwei Cao, Tsinghua University and Tsinghua National Laboratory for Information Science and Technology, China
Yisheng Zhong, Tsinghua University and Tsinghua National Laboratory for Information Science and Technology, China
Lianchen Liu, Tsinghua University and Tsinghua National Laboratory for Information Science and Technology, China
Cheng Wu, Tsinghua University and Tsinghua National Laboratory for Information Science and Technology, China

ABSTRACT

Fine-grained allocation of compute resources, in terms of configurable clock speed of virtual machines, is essential for processing efficiency and resource utilization of data streaming applications. For a data streaming application, its processing speed is expected to approach the allocated bandwidth as much as possible. Automatic control technology is a feasible solution, but the plant model is hard to be derived. In relation to the model-free characteristic, a fuzzy logic controller is designed with several simple yet robust rules. Performance of this controller is verified to outperform classic controllers in response rapidness and less oscillation. An empirical formula on tuning an essential parameter is obtained to achieve better performance.

Keywords: Data Streaming, Fine-Grained Allocation, Fuzzy Control, Grid, Resource Allocation

1. INTRODUCTION

Grid (Foster & Kesselman, 1998) is now playing a major role in providing on-demand resources to various scientific and engineering applications, among which those with data streaming characteristics are gaining popularity recently.

DOI: 10.4018/jghpc.2010100101

Such applications, called grid data streaming applications, require the combination of bandwidth sufficiency, adequate storage and computing capacity to guarantee smooth and high-efficiency processing, making them different from other batch-oriented ones. A case in point is LIGO (Laser Interferometer Gravitational-wave Observatory) (Deelman & Kesselman, 2002), which is generating 1TB scientific data.
per day and trying to benefit from processing capabilities provided by the Open Science Grid (OSG) (Pordes, 2004). Since most OSG sites are CPU-rich but storage-limited with no LIGO data available, data streaming supports are required to utilize OSG CPU resources. Such applications are novel in that (1) they are continuous and long running in nature; (2) they require efficient data transfer from/to distributed sources/sinks in an end-user-pulling way; (3) it is often not feasible to store all the data in entirety because of limited storage and high volumes of data to be processed; (4) they need to make efficient use of high performance computing (HPC) resources to carry out compute-intensive tasks in a timely manner. Great challenge is proposed to provide sufficient resources, including compute, storage and bandwidth to such streaming applications so that they can meet their service level objectives (SLOs) while maintaining high resource utilization.

Just like other grid applications, resource allocation is essential to achieve high efficiency of data processing for streaming applications. But different from the conventional batch-oriented applications, processing efficiency of data streaming applications is co-determined by compute capacity, bandwidth to supply data in real time and storage. Just as proven in our previous work (Zhang & Cao, 2008), compute, bandwidth and storage must be allocated in a cooperative and integrated way. But at that time, emphasis was laid on allocation of bandwidth and storage. As for compute resources, they were just allocated in a coarse-grained way, i.e., each application was assigned to a processor exclusively, which may cause waste of compute capacity for the limitation of data supply speed. In some cases, end users must pay for the compute resources they occupy even if they cannot make full utilization of them. So, it is desirable to allocate fine-grained compute resources for each application, i.e., to allocate just enough compute resources to guarantee smooth processing. Compute resources should also be assigned on demand, and unilateral redundancy of them makes no sense, only to waste users’ budget.

Owe to the progress of virtualization technology, it is possible to allocate fine-grained compute resources. But the premise is to determine the required compute resources according to the needed computing capacity. Unfortunately, it is not so easy for the relationship between the amount of compute resources and the generated compute capacity for a given application is complex because of other influencing factors and it is hard, if not impossible to be obtained. Or put it another way, the precise model is unavailable. It is natural to resort to classical control theory to solve such a tracking or regulation problem as has been done in computing field, but for the absence of precise models, the classical controllers are just baffled. Fortunately, fuzzy logic control theory provides an alternative which requires not the precise models but only some experiences of human beings. In this paper, a fuzzy logic controller (FLC) is designed with some simple but robust fuzzy rules to decide the amount of compute resources for the expected computing capacity, so as to realize the fine-grained compute resource allocation for data streaming applications, which will guarantee service level agreements (SLAs) while maintaining high resource utilization.

The rest of this paper is organized as following: Section 2 formulates an optimization problem and proposes the necessity of fine-grained compute resources allocation, which is resolved with fuzzy controller described in Section 3. Some experimental results are provided in Section 4, to justify the fuzzy allocation. The next section overviews the related research in this field and this paper is concluded in the last section.

2. PROBLEM FORMULATION

In a data streaming scenario, data in remote sources will be transferred to local storage, read by processing program one tuple by another and deleted. From a macroscopic viewpoint, data is just processed in a form of tuple streams.
Related Content

A Waste Management Robot System: Its Implementation and Experimental Results
[www.irma-international.org/article/a-waste-management-robot-system/126174/](www.irma-international.org/article/a-waste-management-robot-system/126174/)

Gridifying Neuroscientific Pipelines: A SOA Recipe and Experience from the neuGRID Project
[www.irma-international.org/chapter/gridifying-neuroscientific-pipelines/45562/](www.irma-international.org/chapter/gridifying-neuroscientific-pipelines/45562/)

A Comparative Study and Algorithmic Analysis of Workflow Decomposition in Distributed Systems

Design and Application of a Containerized Hybrid Transaction Processing and Data Analysis Framework

Migrating Android Applications to the Cloud
[www.irma-international.org/chapter/migrating-android-applications-cloud/69043/](www.irma-international.org/chapter/migrating-android-applications-cloud/69043/)